

AFFTC-PA-05198



# **APPLICATION OF MATLAB®-BASED AUTOMATED TURBINE ENGINE ANALYSIS**

Lt Khoi Ta)

**AIR FORCE FLIGHT TEST CENTER  
EDWARDS AFB, CA**

**September 2005**

**A  
F  
F  
T  
C**

**Approved for public release: distribution is unlimited.**

**AIR FORCE FLIGHT TEST CENTER  
EDWARDS AIR FORCE BASE, CALIFORNIA  
AIR FORCE MATERIEL COMMAND  
UNITED STATES AIR FORCE**



**REPORT DOCUMENTATION PAGE**Form Approved  
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

<b>1. REPORT DATE (DD-MM-YYYY)</b> September 2005		<b>2. REPORT TYPE</b> Technical Paper		<b>3. DATES COVERED (From - To)</b> August 2004 – July 2005	
<b>4. TITLE AND SUBTITLE</b> Application of MATLAB®-Based Automated Turbine Engine Analysis				<b>5a. CONTRACT NUMBER</b>	
				<b>5b. GRANT NUMBER</b>	
				<b>5c. PROGRAM ELEMENT NUMBER</b>	
<b>6. AUTHOR(S)</b> Lt Khoi Ta				<b>5d. PROJECT NUMBER</b>	
				<b>5e. TASK NUMBER</b>	
				<b>5f. WORK UNIT NUMBER</b>	
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES)</b>  Air Force Flight Test Center Edwards AFB CA 93524-6001				<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>  AFFTC-PA-05198	
<b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> 416 FLTS 118 N Wolfe Ave Bldg 1643 Edwards AFB CA 93524				<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>  N/A	
				<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>	
<b>12. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release: distribution is unlimited.					
<b>13. SUPPLEMENTARY NOTES</b> CA: Air Force Flight Test Center Edwards AFB CA                      CC: 012100					
<b>14. ABSTRACT</b> <p>The flight testing of propulsion systems has always centered on engine response and operability. The flight testing and evaluation process can be data intensive due to the sophistication of the hardware and software and the fact that response and operability needs to be verified throughout the entire aircraft flight envelope. This is true whether the item under test is a new engine or a modification of an existing engine. The analysis of engine operability has traditionally been a manually intensive process. Determination of when a critical event occurred (e.g., combustor light-off or engine stability) can vary from engineer to engineer. The variance can be attributed to the fact that critical events are not always clear and at times depend on a "judgment call" by the engineer doing the analysis.</p> <p>This paper will present the automated analysis of engine airtasks and throttle transients. This analysis was performed using a MATLAB®-based analysis and plotting software tool developed at the Air Force Flight Test Center (AFFTC), Edwards AFB, California. The program was called the Airstart and Transient Analysis Program (AirTrans AP). This version of software was planned to be used at AFFTC, Arnold Engineering Development Center (AEDC), and NAVAIR Patuxent River. The AirTrans AP was used on the recent F-15E/F100-PW-229 Digital Electronic Engine Control (DEEC) Group VI project as well as the C-17/F117 electronic engine control Software Control Number (SCN)-9 project. Future plans include use during propulsion testing on the F-35 aircraft.</p> <p>MATLAB® was chosen as the coding language for several reasons. First was the widespread use of MATLAB® throughout industry and academia. This had the potential to allow for a wider group of people to collaborate on the AP code. Second, MATLAB® was easier to code than other programming languages, since it had many pre-existing functions, such as data filtering, plotting, and the ability to input and output in a variety of formats. Finally, an additional strength of MATLAB® was the ability to compile the code and run in a stand-alone mode that did not require the MATLAB® software.</p> <p>This paper will briefly discuss the AFFTC's decisions to use MATLAB® and to automate engine analysis procedures for engine testing. A detailed discussion of the features, user interface, and output of the AirTrans AP will follow. This output includes plots, a list of critical events, and a database. Specifically, this paper will examine the similarities and differences between the automated and manual analysis processes. Analysis from both airtasks and throttle transients will be presented. A final discussion of lessons learned will conclude the paper.</p>					
<b>15. SUBJECT TERMS</b> Engine Airtasks      Airstart and Transient Analysis Program (AirTrans AP)      Flame-out      Hot Start Throttle Transients      Automated Engine Analysis      Blow-out      Cut-off      Hung Start      Light-off					
<b>16. SECURITY CLASSIFICATION OF:</b> <b>Unclassified</b>			<b>17. LIMITATION OF ABSTRACT</b>  None	<b>18. NUMBER OF PAGES</b>  22	<b>19a. NAME OF RESPONSIBLE PERSON</b> 412 TW/EN (Tech Pubs)
<b>a. REPORT</b> Unclassified	<b>b. ABSTRACT</b> Unclassified	<b>c. THIS PAGE</b> Unclassified			<b>19b. TELEPHONE NUMBER (include area code)</b> 661-275-9581



# APPLICATION OF MATLAB®-BASED AUTOMATED TURBINE ENGINE ANALYSIS

1<sup>st</sup> Lt Khoi D. Ta, Propulsion Engineer  
416th Flight Test Squadron  
Bldg 1643, 118 N. Wolfe Ave.  
Air Force Flight Test Center  
Edwards AFB, CA 93534  
Commercial Phone: 661-277-6790  
Email: [khoi.ta@edwards.af.mil](mailto:khoi.ta@edwards.af.mil)

## ABSTRACT

The flight testing of propulsion systems has always centered on engine response and operability. The flight testing and evaluation process can be data intensive due to the sophistication of the hardware and software and the fact that response and operability needs to be verified throughout the entire aircraft flight envelope. This is true whether the item under test is a new engine or a modification of an existing engine. The analysis of engine operability has traditionally been a manually intensive process. Determination of when a critical event occurred (e.g., combustor light-off or engine stability) can vary from engineer to engineer. The variance can be attributed to the fact that critical events are not always clear and at times depend on a "judgment call" by the engineer doing the analysis.

This paper will present the automated analysis of engine airtasks and throttle transients. This analysis was performed using a MATLAB®-based analysis and plotting software tool developed at the Air Force Flight Test Center (AFFTC), Edwards AFB, California. The program was called the Airstart and Transient Analysis Program (AirTrans AP). This version of software was planned to be used at AFFTC, Arnold Engineering Development Center (AEDC), and NAVAIR Patuxent River. The AirTrans AP was used on the recent F-15E/F100-PW-229 Digital Electronic Engine Control (DEEC) Group VI project as well as the C-17/F117 electronic engine control Software Control Number (SCN)-9 project. Future plans include use during propulsion testing on the F-35 aircraft.

MATLAB® was chosen as the coding language for several reasons. First was the widespread use of MATLAB® throughout industry and academia. This had the potential to allow for a wider group of people to collaborate on the AP code. Second, MATLAB® was easier to code than other programming languages, since it had many pre-existing functions, such as data filtering, plotting, and the ability to input and output in a variety of formats. Finally, an additional strength of MATLAB® was the ability to compile the code and run in a stand-alone mode that did not require the MATLAB® software.

This paper will briefly discuss the AFFTC's decisions to use MATLAB® and to automate engine analysis procedures for engine testing. A detailed discussion of the features, user interface, and output of the AirTrans AP will follow. This output includes plots, a list of critical events, and a database. Specifically, this paper will examine the similarities and differences between the automated and manual analysis processes. Analysis from both airtasks and throttle transients will be presented. A final discussion of lessons learned will conclude the paper.



## NOMENCLATURE

**Blow-out** – When the flame from the afterburner or combustor has extinguished. Usually refers to the afterburner.

**Cut-off** – Throttle setting denoting engine off.

**First Rotation** – The first time an engine core accelerates after light-off, refers to a ground measurement.

**Flame-out** – Combustor combustion has extinguished.

**Hot Start** – An airstart where FTIT has exceeded a predefined limit.

**Hung Start** – An airstart in which core speed does not accelerate or decelerate for a certain period of time.

**Idle** – For normal operations, this is the throttle setting for minimum thrust or power. For airstarts, this refers to when the engine has reached an operable level and combustor combustion can be maintained.

**Initiation** – A command from one throttle setting to another.

**Light-off** – The start of combustor combustion or afterburner ignition.

**Military Power** – Maximum thrust without the use of afterburners.

**Maximum Power** – Maximum power with afterburners.

**N1** – Fan speed measured in RPM.

**N2** – Core speed measured in RPM.

**N2%** – Core speed measure in percent of maximum RPM.

**No-Light** – Failure of the afterburner to ignite after a certain time period.

**PB** – Combustor pressure.

**Relight** – Restart of combustor combustion or afterburner ignition after a blow-out.

**Stall** – Disruption of airflow across the blades the engine fan or core stages.



## INTRODUCTION

The Airstart and Transient Analysis Program (AirTrans AP) was an application to automate the initial analysis of turbine engine airstart and throttle transient test data. The Air Force Flight Test Center (AFFTC), Edwards AFB, California, pursued the development of the AirTrans AP for several reasons. The application will someday standardize the analysis of engine airstarts and throttle transients across the Center. The AirTrans AP will not only standardize the tools for analysis, but also databasing. It also has the potential to gain use with Arnold Engineering Development Center (AEDC), Arnold AFB, Tennessee, and Naval Air Systems Command (NAVAIR), Patuxent River, Maryland. By using MATLAB® as a development tool, the three centers have agreed to work together to develop and improve the AirTrans AP. This will allow for the mutual use of a single analysis tool for engine development testing. Although the AirTrans AP was developed using MATLAB®, it could also be compiled as a standalone program which did not require MATLAB® to run.

The central idea behind the AirTrans application was to develop an automated and standardized way to analyze, plot, list, and database critical events from turbine engine airstart and throttle transient testing. Traditionally, the analysis of engine response and operability has been a manual and time intensive process. The plotting of results for reporting has usually been a separate process. The AirTrans AP incorporates analysis and plotting into a single process. A list and database of parameters at key events has also been added. Listing allows for the immediate review of selected parameters at critical events. Databasing allows for the sorting and querying of data, which then allows for the comparison to previous test data. The automation of analysis, plotting, listing, and databasing greatly speeds and standardizes the process of engine operability analysis. Although the application was still in the development phase, it was well on the way to becoming a complete tool for the analysis of engine response and operability.

This paper discusses the results from running generic engine test data through the AirTrans AP. The results from the application were compared to the traditional and established methods of engine operability analysis. Airstarts and throttle transients were first analyzed using manual methods; the analysis was then repeated with the AirTrans AP. This comparison was done to see if the application could obtain results similar to traditional methods of analysis. The test points used in the comparison were airstarts and augmented and nonaugmented throttle transients. The features and capabilities of the AirTrans AP are discussed, as well as the similarities and differences to traditional methods.

## TRADITIONAL OPERABILITY ANALYSIS

The key part of engine operability analysis is identifying critical events. Some events are easily identified, such as the increase in combustor pressure when an engine lights-off during an airstart. Other events, such as engine stability, are much more subjective and must be evaluated to ensure repeatability and consistency. The following is a discussion of the critical events and their corresponding products of analysis. It is offered as a background on traditional analysis of engine response and operability.

### Airstart Analysis



Airstarts have four key events: cutoff, initiation, lightoff, and idle. Cutoff was the pilot throttle (PLA) command to shut the engine off. Initiation was the pilot command to restart. Lightoff was when fuel in the combustor has ignited. Lightoff was determined from the rise in combustor pressure (PB) immediately after combustion has occurred or by the sudden jump in combustor discharge temperature (FTIT) or an increase in compressor rotor speed (N2). However, the rise in combustor pressure is the preferred method since it is the most immediate. Turbine temperature and core speed have to overcome a degree of thermal and rotational inertia, while air pressure is faster since there is no appreciable inertia. Figure 1 presents the events that occur during a typical airstart. The red circle in figure 1 shows the rise in combustor pressure. Idle is when the engine has reached stable idle thrust. For some engines, idle is achieved when combustor pressure reaches a certain point. For other engines, the idle point varies with flight conditions and can be subjective. The two products which typically quantify start performance are time-to-light and time-to-idle.

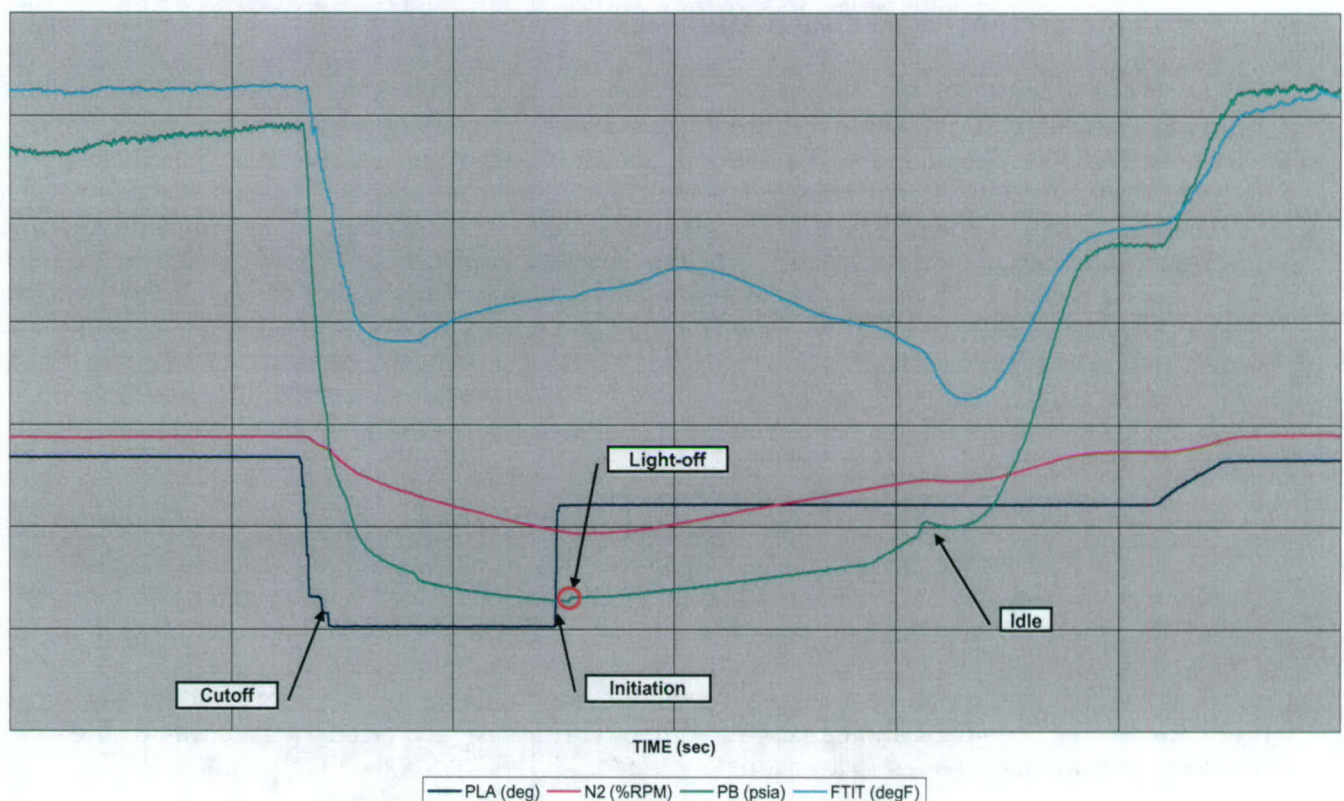


Figure 1 Typical Engine Airstart

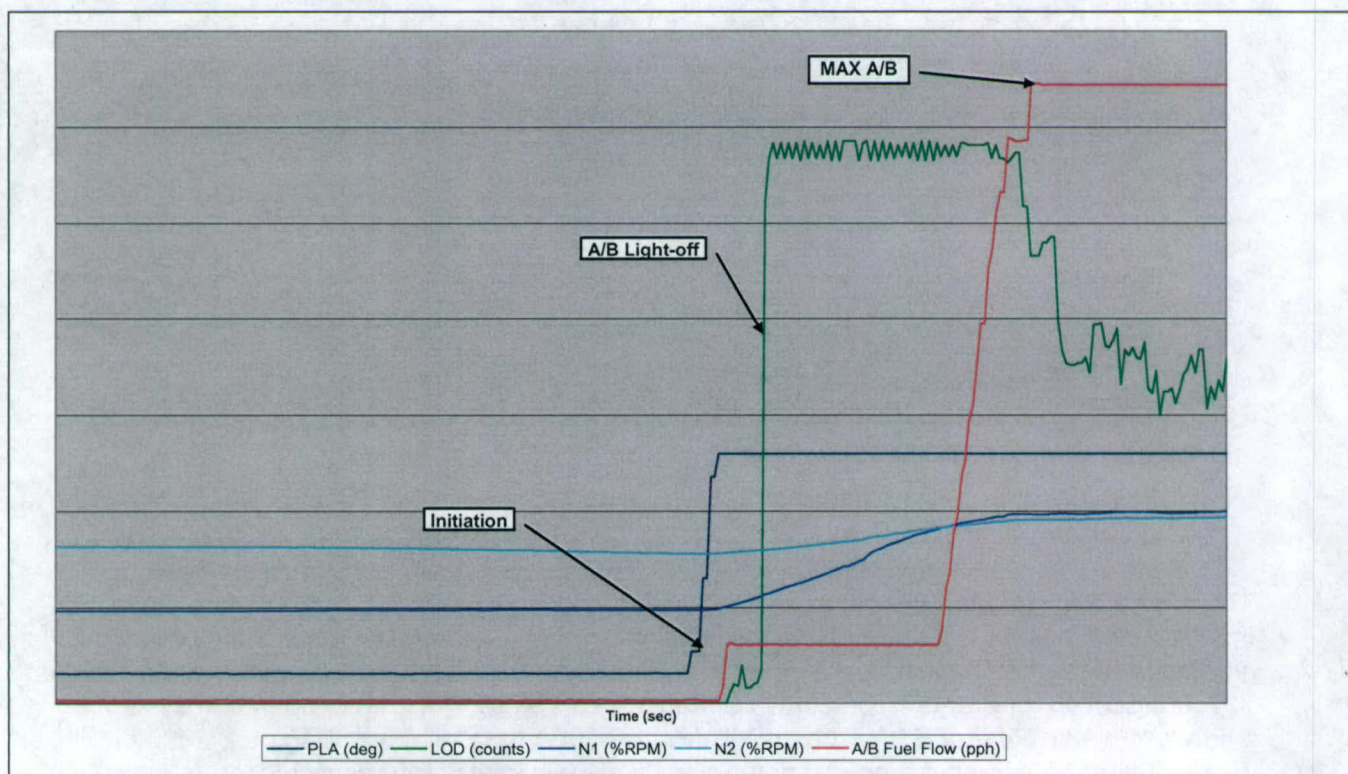
### Throttle Transient Analysis

Throttle transients can be divided into augmented and nonaugmented transients. Both produce different products of analysis. For augmented transients the products of analysis are time-to-light and time-to-max. Time-to-light refers to the time to achieve afterburner ignition. Time-to-Max refers to the time to achieve maximum stable afterburner thrust. For a typical fighter engine maximum afterburner available varies with conditions. For a nonaugmented transient, the products of analysis are time to stable engine and time-to-98-percent thrust change. Both



apply to engine accelerations and decelerations. Stable core speed at military power varies with flight conditions and can be subjective to determine. The time-to-98-percent thrust change can be determined by using the thrust calculated by an onboard thrust model, or, when a thrust model is not available, time-to-98-percent thrust change is measured as the time-to-98-percent of stable fan speed. The second method of 98 percent determination is usually much clearer since it is easier to analyze when the fan speed has reached a certain point rather than when fan speed has stabilized.

Figure 2 presents a typical augmented transient. The initiation point (on dark blue line) is the snap transient to maximum power, which in this case is a snap from idle. Augmented transients can also start with throttle snaps from military or part power. After afterburner initiation, the lightoff detector (LOD) will detect the afterburner flame. Time from initiation to the sharp spike in LOD counts (green line) is the time-to-light.



**Figure 2 Snap Idle-to-Max Augmented Throttle Transient**

Figure 3 (page 6) presents a typical nonaugmented transient. The initiation point is the throttle snap from idle to military power. For engine accelerations, time to stable engine is called time-to-mil. Time-to-98-percent mil is determined by taking the average of fan speed (N1) once the engine has stabilized and finding the time that N1 has reached 98 percent of that average. This avoids the problem of determining when the fan speed has stabilized. Neither idle nor military power has a distinctive marker – both are subjective determinations.

The analysis of engine operability has always been a time consuming and laborious process. Each test point usually takes anywhere from five to fifteen minutes to analyze. This process includes reading in the raw data, applying needed unit conversions, plotting parameters, finding location of key events, and determining products of analysis. Taken singularly this is not a long time. However, taken over the course of an entire flight test project, a complete analysis can



take days if not weeks. It is not uncommon for a typical engine flight test project to have several hundred test points. This is compounded by the subjective nature of part of the analysis, which must be done carefully to ensure consistency and repeatability.

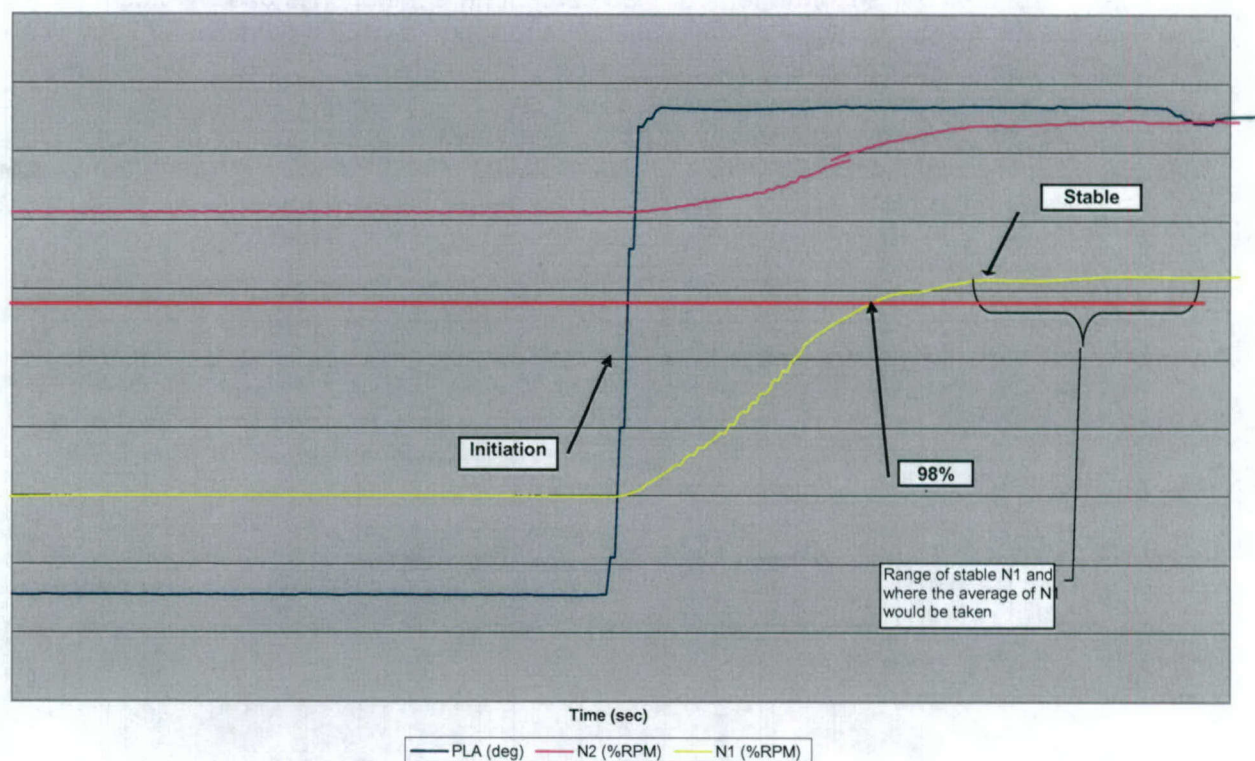


Figure 3 Snap Idle-to-Mil Nonaugmented Throttle Transient

### THE AIRTRANS AERO-PROPULSION APPLICATION

The AirTrans AP automated the process of engine operability analysis. As a computer-based analysis tool, the application could be easily used by a variety of users, testing cargo, bomber, and fighter aircraft. The hardware requirement was a desktop personal computer with a Pentium III 750 processor or greater. The software requirements were Windows™ operating system, Adobe Acrobat® Reader 5.0, Microsoft™ Access® 2000, and Microsoft™ Excel® 2000 or later. The AirTrans AP did more than just determine key events. Its features included:

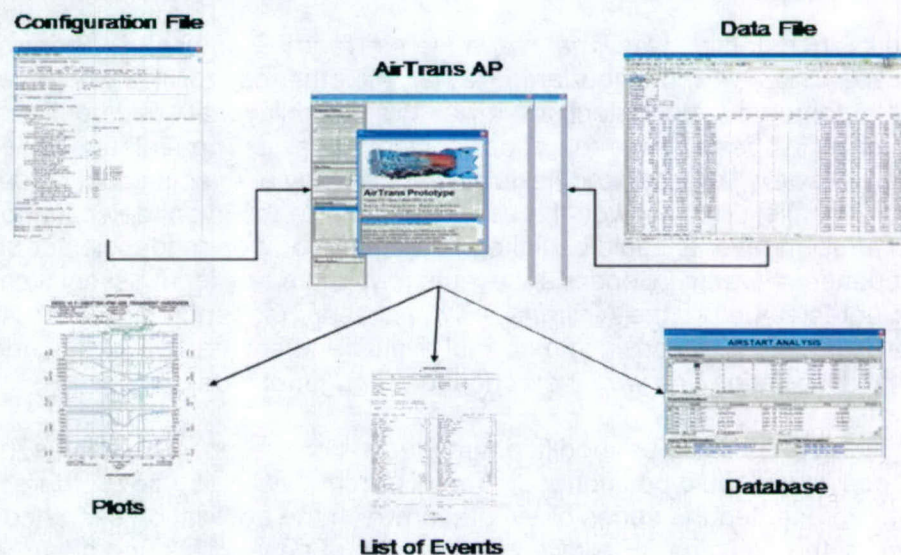
1. **Consistent and Repeatable Analysis** – The application used a configuration file that defines, among the other processes of the application, how events were to be detected. This ensured that similar events were analyzed exactly the same way from test point to test point. This was especially important for subjective events.
2. **Data Input** – As data was read, engineering units and data nomenclature were automatically assigned for all parameters.
3. **Plotting** – The format and number of plots were defined by the user and was consistently repeated for each test point. This was automatically done as part of the analysis.



4. **Listing** – A list of the results from selected parameters was automatically generated for all detected events. This allowed for a quick review of data to ensure that the events are detected correctly.
5. **Databasing** – All detected events and selected parameters could be sent to a database. In addition, hyperlinks to associated plots, lists, raw data, and configuration setup files were also sent to the database. This allowed for the quick collection of data in a format that could be sorted and queried. The database also allowed for easy comparison of test results between projects.

### The Structure of the AirTrans AP

Figure 4 shows the structure of the AirTrans AP. There were two input files, a data and configuration file. From these two input files, data would be sent to plots, a list of events, and a



database.

**Figure 4 Flow Chart of the Structure of the AirTrans AP**

The configuration file contained sections that the AirTrans AP used in its analysis. The following sections are part of the configuration file:

1. **Program Configuration** – Contained administrative information, including project name, classification level, test engine position (for multiengine aircraft), and default directories for AP output.
2. **Module Configuration** – Contained information on the type of formats that the application could read and write. The formats include; AFFTC Comma Separated Values (CSV), AEDC Parameter-Oriented Data (POD), and Lockheed Martin Hierarchical Data Format (HDF) 4 files.
3. **Engine Configuration** – Specified the name and the number of engines.
4. **Parameter Configuration** – Contained information on the parameters that will be used for analysis. This section told the application what was the source and the units of the data. It also told the application to which engine the data pertains. If needed,



instructions could be added in this section to perform basic mathematic operations or automatically compute the derivatives of the data being read.

5. **Event Configuration** – Specified which events to look for and what were the conditions for the events. This was the most important section of the configuration file. This was where the user told the application how the analysis should be performed.
6. **Plot Configuration** – Specified which parameters and how they would be plotted.
7. **List Configuration** – Specified which parameters to output to a list for each detected event.
8. **Database Configuration** – Specified which parameter should be sent to the database when an event was detected.

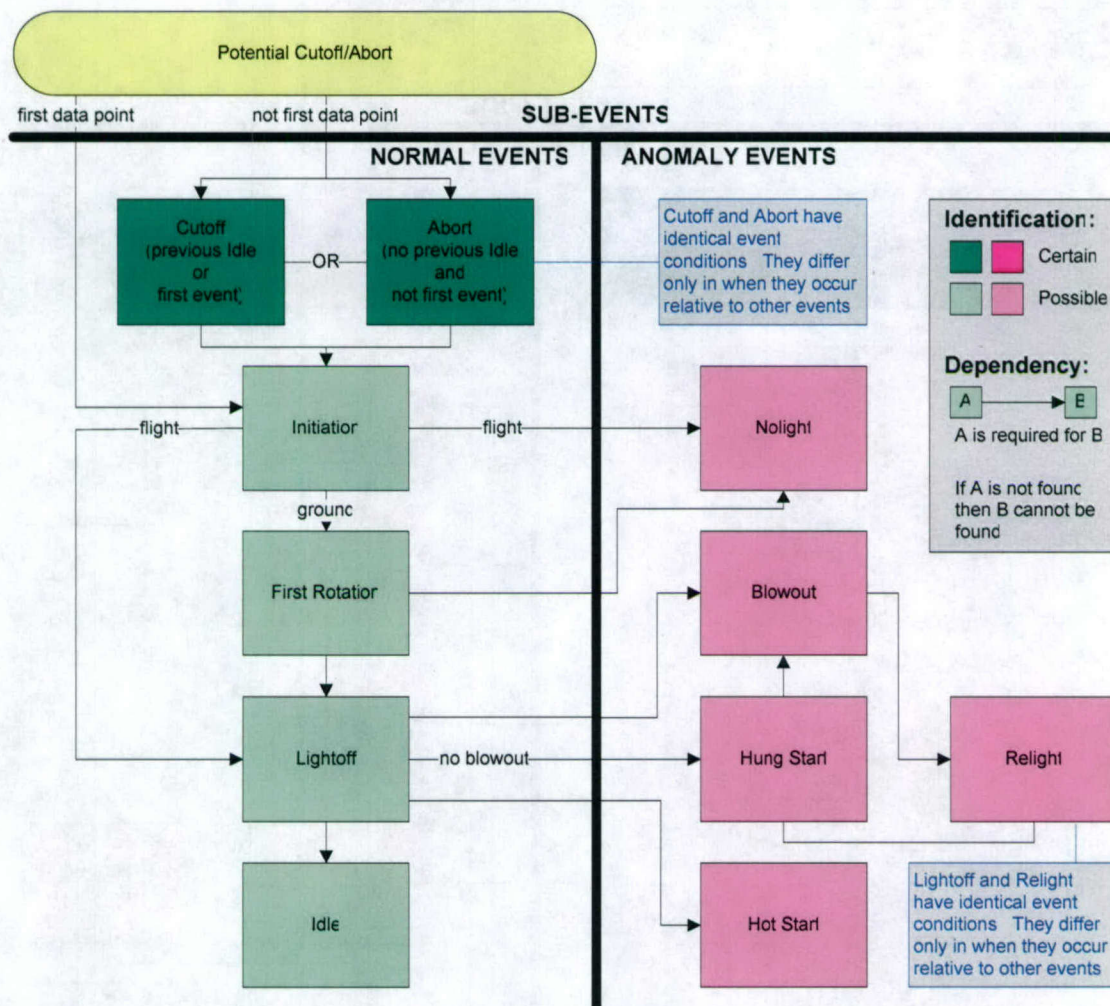
Users of the application spend most their time editing the configuration file. It is through this file that the user sends instructions on how to perform the analysis.

### **The Process of Event Detection**

Automated event detection was what made the AirTrans AP so useful. That is not to say that the other capabilities were unimportant. Rather, the other capabilities would be of little use without the automated event detection. It was this capability that sped up the task of analysis and could assist in speeding the report generating process. The AirTrans AP event detection capabilities were very versatile and flexible. A wide array of events could be defined through the configuration file. This allowed the user to customize the event detection for any particular engine. In addition, events could be linked together to form dependencies or be decoupled and left as independent events. Dependent events required a relational association; one event must have been detected before the AirTrans AP would search for another event. These dependent events were further broken down into normal events and anomalies. Independent events had no relational association and were searched for throughout a data series.

The event hierarchies could be modified at the user's discretion. The names, hierarchy order, and event conditions could be changed. The hierarchy could also be reordered. The hierarchies represented the series of events for which the application searched. In addition, the open nature of the AirTrans AP allowed the creation of new events and hierarchies. Figure 5 presents the airstart event hierarchy; this is the normal sequence of events for almost all airstarts.

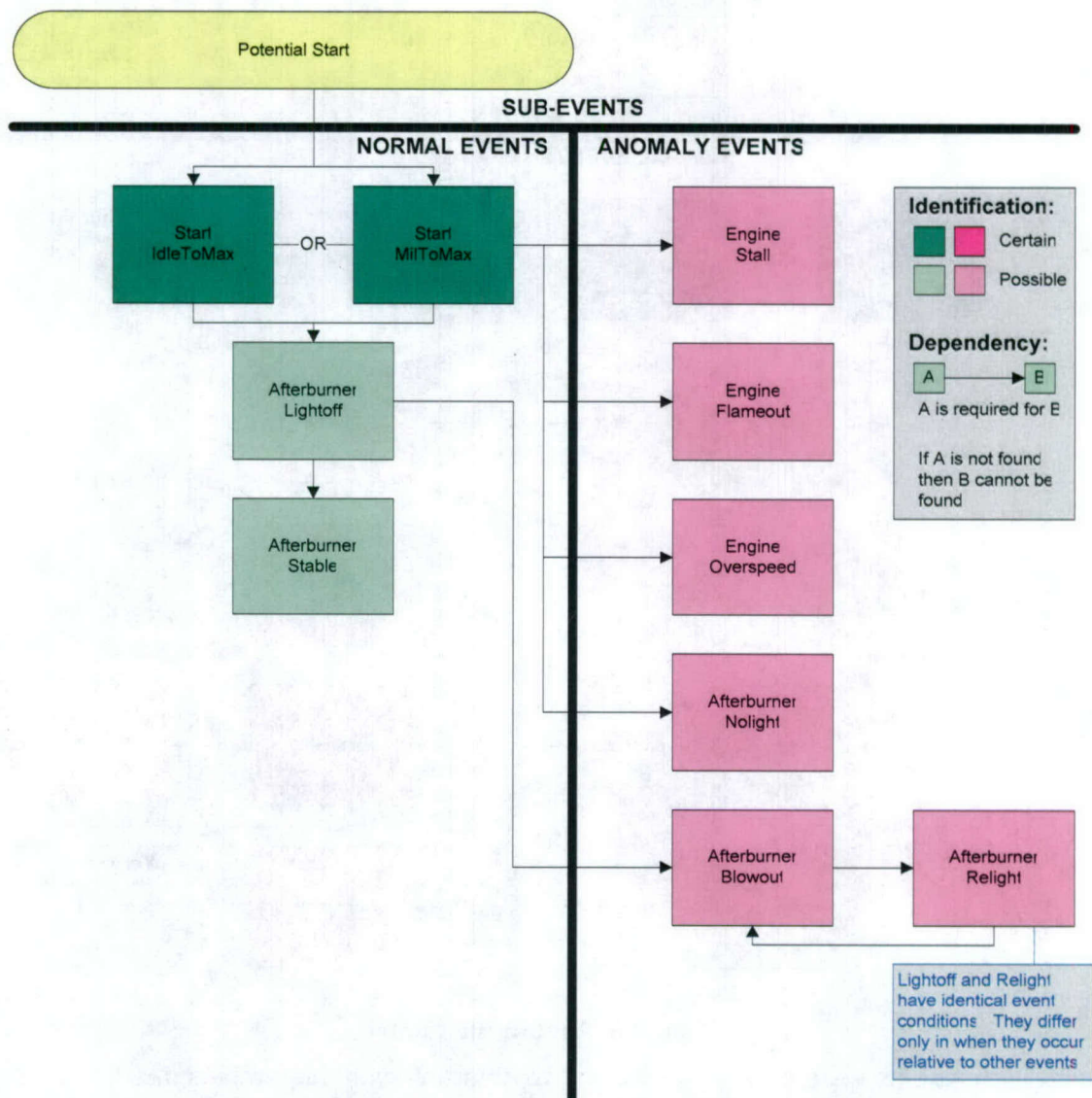




**Figure 5 Airstart Hierarchy**

Augmented transients were broken down into two different event hierarchies due to the different events involved for afterburner lightoff and cancellation. Figure 6 presents the event hierarchy for afterburner light-offs. The AP was configured to detected snap idle-to-max or mil-to-max transients to initiate the afterburner lightoff event hierarchy. Figure 7 (page 10) presents the event hierarchy for afterburner cancellations. Like afterburner initiation, there were two possible event triggers for afterburner cancellation. These triggers were max-to-mil and max-to-idle snap transients. The exact method the application used to detect initiation is discussed in the results of analysis section.





**Figure 6 Afterburner Initiation Throttle Transient Event Hierarchy**

Figure 8 (also page 10) presents the event hierarchy for nonaugmented transients. Initiation was a throttle snap from idle-to-mil or mil-to-idle. The sequence of events was similar for both the afterburner cancellation and military transient hierarchies. All normal and anomaly events were the same with one exception: the afterburner cancellation event hierarchy searched for the cancellation event. The AirTrans AP distinguished between the two by their initiation events. 98 percent stable was ignored for max-to-mil transients. N1 usually did not change much from maximum to military power.



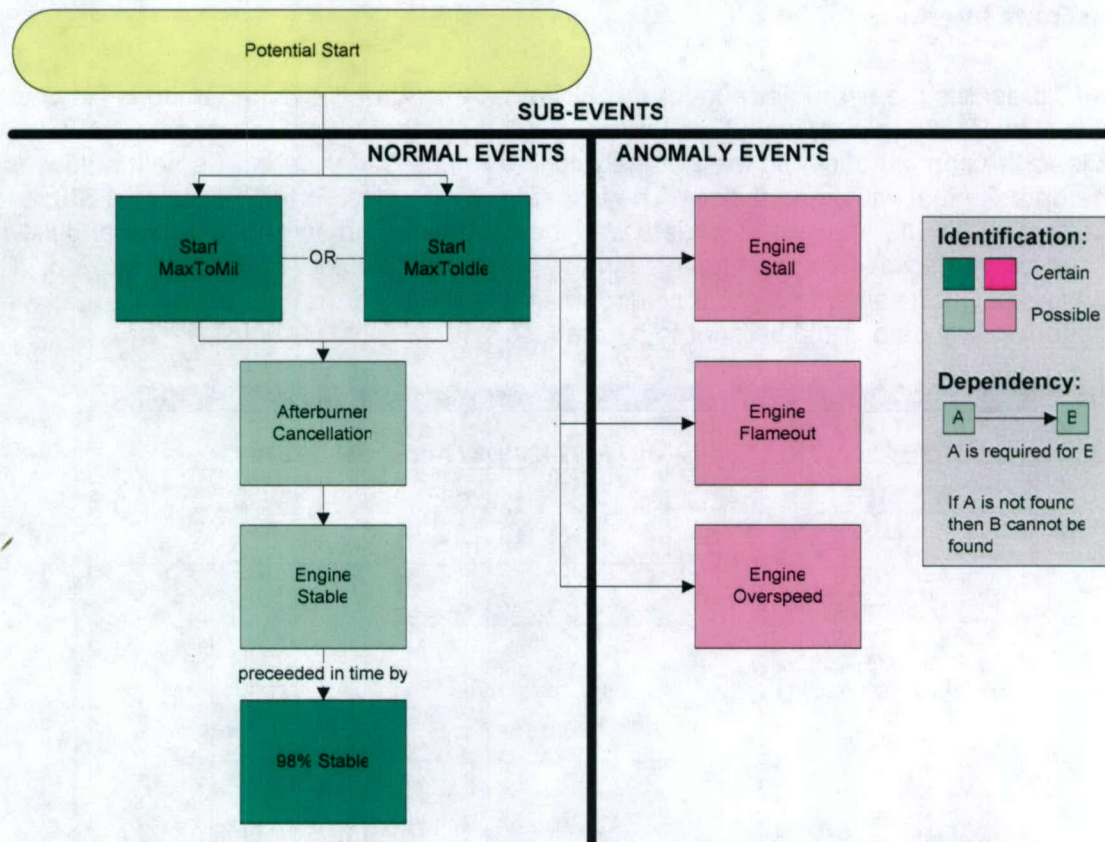


Figure 7 Afterburner Cancellation Throttle Transient Event Hierarchy

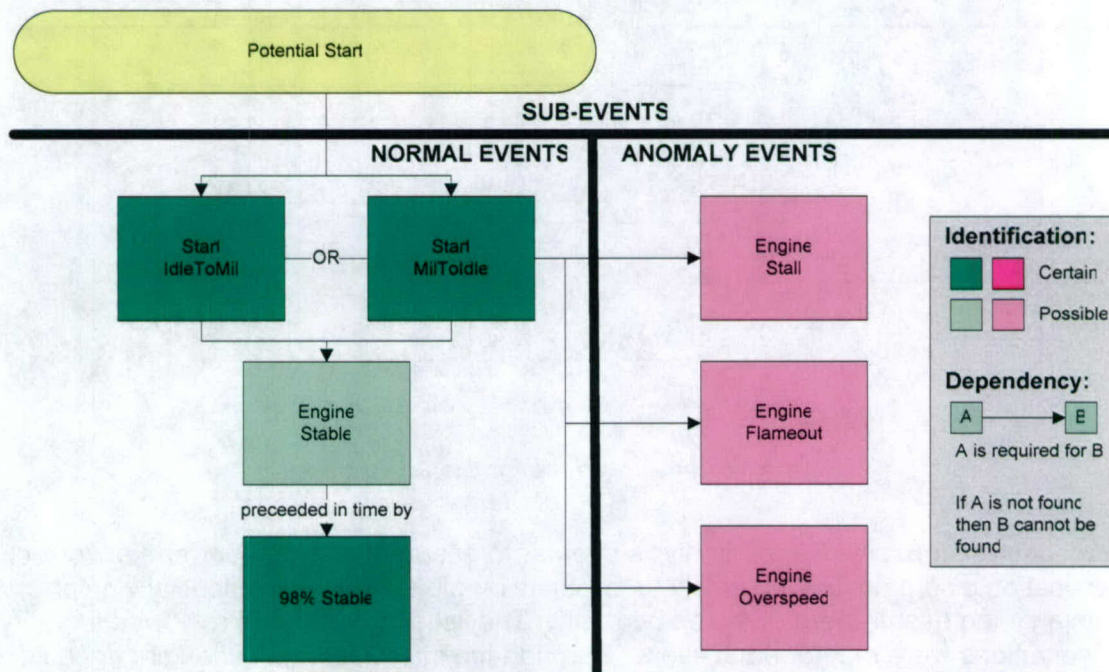


Figure 8 Military Transient Event Hierarchy



## The AirTrans Interface

Figure 12 presents the user interface for the AirTrans AP. After the configurations file was completed, the test engineer would continue to work through the user interface. The first step was to load the configuration file for the analysis that will be performed. The next step was to load the data file that will be analyzed. The last step was to click **Find Events and Show Output**; this begun the analysis. The data was then displayed on screen as a plot and list showing key events, which were displayed in the format that the test engineer specified. The plot and list could be saved to a portable document file (PDF) format. Events and accompanying key data could be sent to a database.

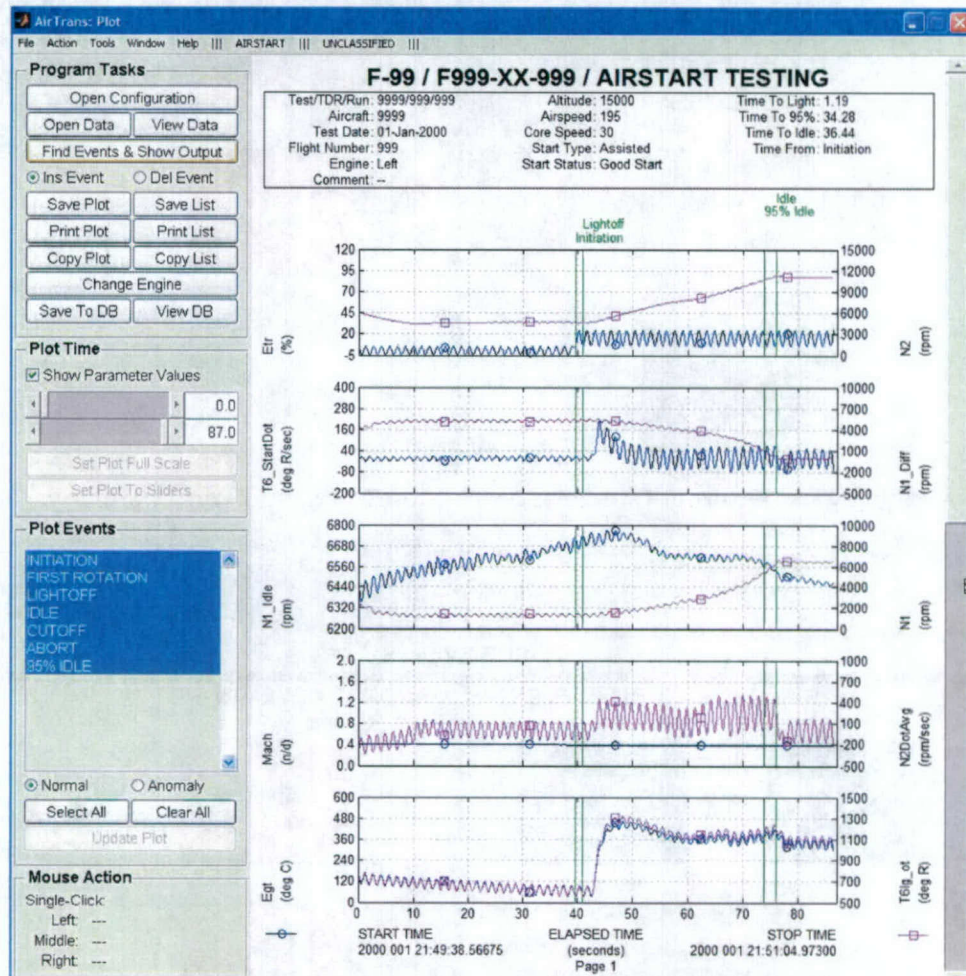


Figure 12 User interface for the AirTrans AP

The test engineer must review the analysis in order to ensure that it was performed correctly. The application also provided the ability to 'zoom in' on plots. This was especially helpful when determining if the events were detected correctly. The list should also be reviewed to verify the proper conditions were met for each event. Through the interface, the test engineer could insert and delete events. The ability to delete events was useful when the application performed analysis that was not needed by the test engineer. There were a few test points where the engine had not fully stabilized but the throttle had already proceeded to next transient. In these cases, the engine was very close to, but had not quite reached, the engine stable condition.



The ability to insert events allowed the test engineer to declare when the conditions were “good enough” for engine stable.

All detected events and selected parameters could be sent to the database. Saved plot and list files were recorded as hyperlinks in the database. The configuration and input data file were also saved as hyperlinks. Figure 13 presents an example database from the AirTrans AP.

**Airstart Analysis**

## AIRSTART ANALYSIS

**Test Information:**

	AircraftNumber	TestNumber	RequestNumber	RunNumber	EngineName	TestDate	FlightNumber	Altitude	Airsp
▶	9999	9999	999	999	LEFT	01-Jan-2000	999	15,000	
*									

Record: 1 of 1

**Event Information:**

	EventName	EventTime	ETR	N1	N1_IDLE	N1DIFF	N2	T6DOT
▶	INITIATION	2000 001 21:50:18.22300	8.33	1,296.14	6,717.68	5,421.54	4,684.97	12.74
	LIGHTOFF	2000 001 21:50:19.41050	6.60	1,180.92	6,739.90	5,558.97	4,679.97	3.90
	95% IDLE	2000 001 21:50:52.50425	13.97	5,405.83	6,567.96	1,162.13	10,701.36	-12.36
	IDLE	2000 001 21:50:54.66050	15.29	6,464.25	6,537.77	73.52	11,257.93	28.78
*								

Record: 1 of 4

**Input File Information:**

Configuration: [airstart\\_config\\_test\\_public.csv.ini](#)

Parameter: [airstart\\_parameter\\_test\\_public.DHMS.csv](#)

**Output File Information:**

Plot: [test.pdf](#)

List: [test\\_list.pdf](#)

Figure 13 Example Airstart Database used by the AirTrans AP

Figure 14 (page 13) presents an example analysis of a sample data file. This example shows the simplicity of the steps after completion of the configuration file. It was easy to see how AirTrans AP saved a great deal of time. In three steps, the process of engine operability analysis was completed. Once the configuration and data files were loaded, the application searched for critical events and computed the different products of analysis.



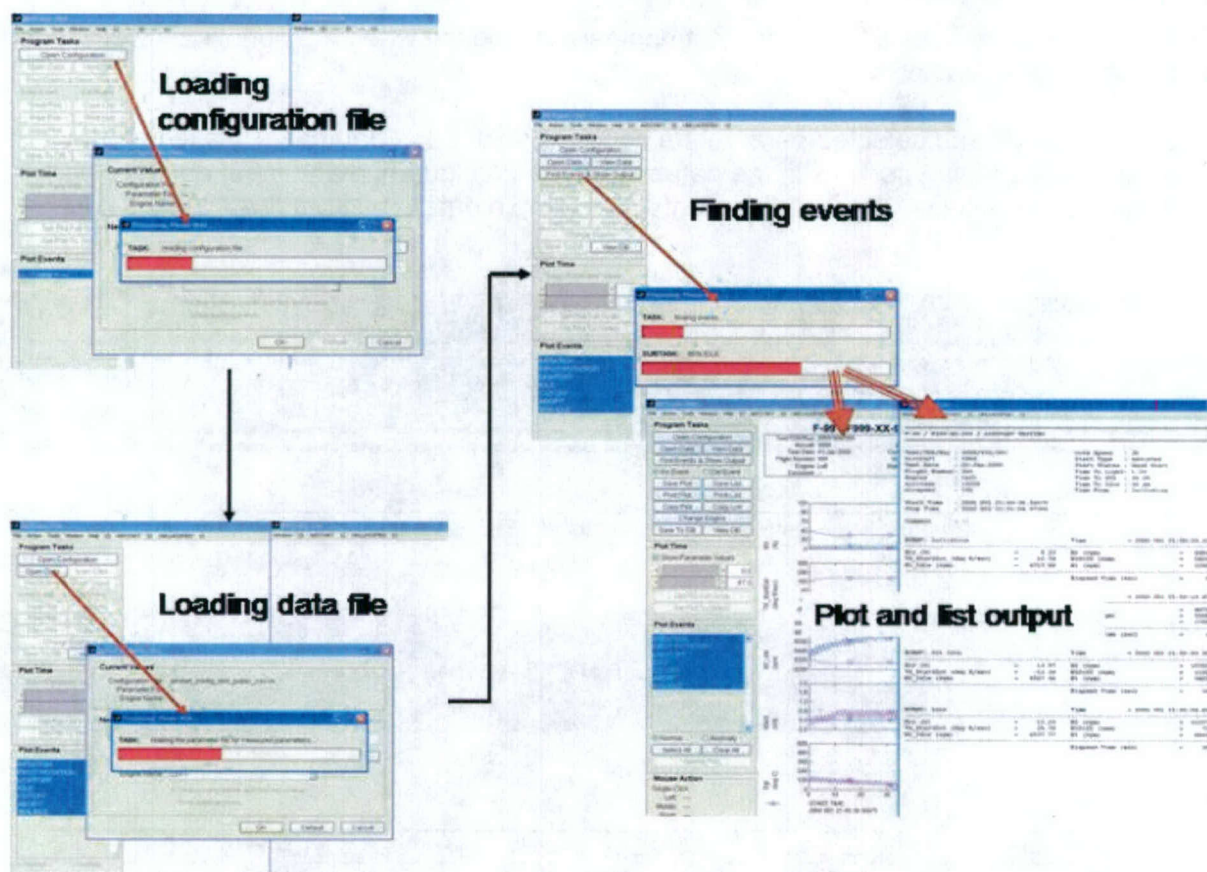


Figure 14 AirTrans AP Example Analysis

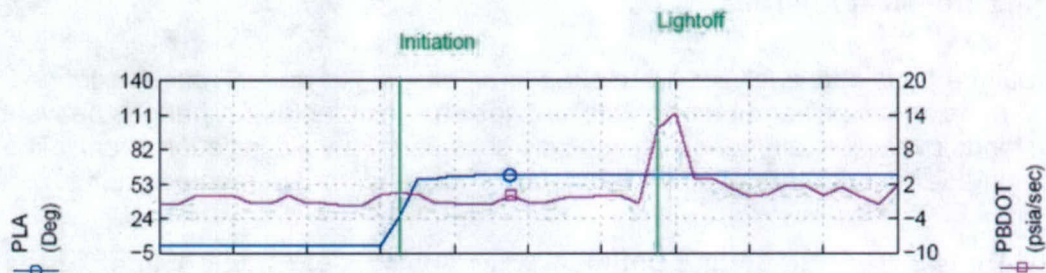
## RESULTS OF ANALYSIS

There are many methods to analyze engine response and operability. The methods and parameters used for analysis vary from engine to engine and from project to project. The following event criteria were chosen for simplicity and ease of comparison. The methods chosen were not necessarily what were being practiced at AFFTC. The goal was to show how well the AirTrans AP compares to manual methods. The manual method was treated as the truth source for all products of analysis. All differences were taken as the difference between the manual and AirTrans AP methods of analysis.

### Airstarts Analysis

The event criterion used for cutoff was a Power Lever Angle (PLA) setting of less than 15 degrees. The opposite event criterion was used for initiation: a PLA setting greater than or equal to 15 degrees. To detect lightoff, the AirTrans AP used the derivative of combustor pressure. The rise in combustor pressure was very abrupt, but also very small at lightoff. The derivative of PB was clear and distinct at lightoff. Figure 15 (page 14) presents an example of the jump in the derivative of burner pressure and the detection of lightoff. In this example, the derivative of PB is between 8 and 14 for the interval at lightoff. The numerical method used to compute the derivative was the unweighted three point difference method.





**Figure 15 Derivative of PB Jumps Distinctly at Lightoff**

A combustor pressure of 50 psia was used as the threshold for idle thrust. For this sample analysis, 50 psia was considered a good general rule of thumb for idle determination. This was used initially as an easy comparison between the manual and AirTrans method of analysis. Table 1 presents a description of the event criteria used for the airstart analysis.

**Table 1 Airstart Analysis Event Criteria**

Analysis	Cutoff	Initiation	Lightoff	Idle
AirTrans	PLA < 15	PLA ≥ 15	PBDOT > 3	PB ≥ 50
Manual	PLA < 15	PLA ≥ 15	Sharp rise in PB shortly after initiation	PB ≥ 50

Table 2 presents the airstart analysis results from both methods. The results were comparable. The time-to-light was 0.05 second faster for all test points analyzed with the AirTrans application. This was attributed to the fact that the derivative of combustor pressure always led the actual change in combustor pressure by one time cycle, due to the derivative calculation technique. As expected, the detection of combustor pressure was clear for determining idle thrust for both the manual and AirTrans AP methods.

**Table 2 Airstart Analysis Results**

Test Point	Aim Conditions			Manual Analysis		AirTrans Analysis		Time Difference	
	Pressure Alt (ft)	Mach or KCAS	N2 (% RPM)	Time to Light (sec)	Time to Idle (sec)	Time to Light (sec)	Time to Idle (sec)	Time to Light (sec)	Time to Idle (sec)
1	Low	Low	30	1.23	40.46	1.18	40.46	0.05	0
2			50	1.53	17.96	1.48	17.96	0.05	0
3			26	4.53	38.60	4.48	38.60	0.05	0
4	Medium	Low	30	3.43	79.04	3.38	79.04	0.05	0
5			50	1.58	34.52	1.53	34.52	0.05	0
6		Medium	30	2.08	26.12	2.03	26.12	0.05	0
7			50	1.58	14.30	1.53	14.30	0.05	0
8	Low	1.43		67.94	1.38	67.94	0.05	0	
9	Medium	Medium		1.48	69.74	1.43	69.74	0.05	0
10	High	1.78		21.02	1.73	21.02	0.05	0	
11	High	Low		2.03	57.86	1.98	57.86	0.05	0
12	Medium		30	1.83	51.98	1.78	51.98	0.05	0
13			50	1.28	18.98	1.23	18.98	0.05	0



## Augmented Transient Analysis

The augmented transients analyzed were snap throttle inputs. The AirTrans AP defined idle, military, and maximum power as the throttle settings shown in table 3. Changes between these throttle settings over a certain period of time was considered a valid initiation event. The same throttle setting definitions were used for both augmented and nonaugmented analysis.

**Table 3 Throttle Setting Definitions for Analysis of Engine Transients**

Idle	Military	Maximum
$14 \leq \text{PLA} \leq 18$	$83 \leq \text{PLA} \leq 91$	$127 \leq \text{PLA} \leq 132$

The use of specific throttle settings allowed for two things. First, it allowed for the determination of which transient to analyze. To ensure that the transient was a valid snap transient, the change from one throttle setting to the next had to be completed within a specified amount of time. Any transient that started or ended at a PLA setting outside the predefined throttle definitions were declared invalid and not analyzed. Second, sometimes transients were not performed perfectly. Pilots sometime overshot or undershot the intended requested position. As long as the initial and final throttle settings were reached in the required time, the application declared the transient a valid snap transient.

The detection of afterburner lightoff and stable afterburner were exactly the same for both methods of analysis. The event criterion for afterburner lightoff was a LOD of 250 counts or greater. For stable afterburner, the event criterion was maximum stable afterburner fuel flow. Maximum afterburner fuel flow varied based on flight condition and was lower at higher altitudes and lower airspeeds. Table 4 presents the event criteria used for augementer analysis.

**Table 4 Augmented Transient Analysis Event Criteria**

Analysis	Initiation	Afterburner Lightoff	Maximum Afterburner
AirTrans	<ul style="list-style-type: none"><li>• Afterburner initiation<ul style="list-style-type: none"><li>◦ Idle to max within 2 seconds</li><li>◦ Mil to max within 2 seconds</li></ul></li><li>• Afterburner cancellation<ul style="list-style-type: none"><li>◦ Max to idle within 2 seconds</li><li>◦ Max to mil within 2 seconds</li></ul></li></ul>	LOD > 250	Afterburner fuel flow has reached the maximum value for flight condition
Manual	PLA $\geq$ greater initial condition, valid only if a step or bodie transient	LOD > 250	Afterburner fuel flow has reached the maximum value for flight condition

Table 5 presents a sample of results for augmented transient analysis. The AirTrans AP was comparable to the manual method of analysis. The differences in the products of analysis were small, less then 0.1 second. These sample points cover a wide spectrum of altitudes and airspeeds. The time differences for the other test points were also comparable.



**Table 5 Augmented Transient Analysis Results**

Test Point	Target Conditions			Manual Results		AirTrans Results		Time Difference	
	Altitude (ft)	Airspeed	Throttle Profile	Time to Lightoff (sec)	Time to Max (sec)	Time to Lightoff (sec)	Time to Max (sec)	Time to Lightoff (sec)	Time to Max (sec)
1	Low	Low	1	1.48	4.88	1.53	4.93	-0.05	-0.05
2			2	1.03	2.43	1.08	2.48	-0.05	-0.05
3		Medium	3	1.43	4.03	1.48	4.08	-0.05	-0.05
4			4	1.03	2.33	1.08	2.38	-0.05	-0.05
5	Medium	Low	1	5.03	7.83	5.08	7.88	-0.05	-0.05
6			2	1.38	4.13	1.28	4.03	0.10	0.10
7		Medium	2	1.13	3.58	1.18	3.63	-0.05	-0.05
7			1	3.48	5.98	3.53	6.03	-0.05	-0.05
8		High	3	3.18	5.98	3.23	6.03	-0.05	-0.05
9			4	1.13	3.03	1.18	3.08	-0.05	-0.05
10	High	Low	2	1.28	5.68	1.33	5.73	-0.05	-0.05
11			4	1.18	5.68	1.23	5.73	-0.05	-0.05

Profiles: 1. Idle-Max  
2. Mil-Max  
3. Max-Idle-Max (Bodie transient)  
4. Max-Mil-Max (Bodie transient)

### Nonaugmented Transient Analysis

Like augmented transients, the only nonaugmented transients analyzed were snap throttle inputs. The determination of initiation was also the same. The AirTrans AP used the same throttle settings for nonaugmented transients as it did for augmented transients. The application was able to distinguish between the two transients, since nonaugmented transients only traveled between idle and military power.

The determination of 98 percent military power was different for the manual and AirTrans AP analysis methods. The manual method took the average of fan speed over a stabilized range and found where the fan speed first exceeded 98 percent of that average. The AirTrans AP determined stable engine by finding where fan speed did not vary by more a specified range and period of time; the fan speed at the beginning of this period was noted. The application then found 98 percent of the difference between initiation and stable engine fan speed; this point was 98 percent stable engine. This method was more consistent than the manual method; however, the conditions were not always met for stable engine. There were a few test points where the AirTrans AP proceeded to the next event before the engine was fully stable which required the analysis engineer to manually insert the stable event. Table 6 presents the event criteria for nonaugmented analysis.

**Table 6 Augmented Transient Analysis Event Criteria**

Analysis	Initiation	98% Mil	Mil (Stable Engine)
AirTrans	Pre-idle to post-mil within 2 seconds	0.98 of $\Delta N1$ between	$\Delta N1 = \pm 50$ RPM,



	Pre-mil to post-idle within 2 seconds	initiation and stable engine	for 1.5 secs
Manual	PLA $\geq$ 50, valid only if a step or bodie transient	N1 $\geq$ 0.98 Stable N2	Average of N1 over a stable range

Table 7 presents a sample of results for nonaugmented analysis. Once again, the results from the two methods of analysis were comparable. The time differences for the test points not included in this sample were also very comparable.

**Table 7 Nonaugmented Transient Analysis Results**

Test Point	Aim Conditions			Manual Results	AirTrans Results	Time Difference
	Altitude (ft)	Airspeed	Throttle Profile	Time to 98% Mil (sec)	Time to 98% Mil (sec)	Time to 98% Mil (sec)
1	Low	Medium	1	4.43	4.48	-0.05
2			2	4.53	4.43	0.10
3		High	2	3.63	3.58	0.05
4	Medium	Low	1	6.03	5.93	0.10
5			2	4.53	4.58	-0.05
6		High	2	5.68	5.68	0.00
7	High	Low	2	3.68	3.83	-0.15

Profiles: 1. Idle-Mil  
2. Mil-Idle-Mil

### Using Different Methods of Analysis

The configuration file made the automation of engine operability analysis a much simpler task. The configuration file also simplified using different methods of analysis, which allowed a test engineer to compare methods. Most digital engine controllers had internal logic to determine when an engine has reached idle. For this project it was close to 50 psia combustor pressure. An internal idle bit was used to declare that an engine was at idle. This bit would go to '1' once the controller determined that the engine had reached idle.

The airstart data was reanalyzed using the idle bit as the determination for idle. All airstart runs were reanalyzed with the AirTrans AP except for a secondary mode airstart. Table 8 (page 18) presents a comparison of these two determinations for idle. The engine did not declare idle at the same time combustor pressure reach 50 psia; however, the differences were all less than 1 second and in most cases less than 0.5 second.

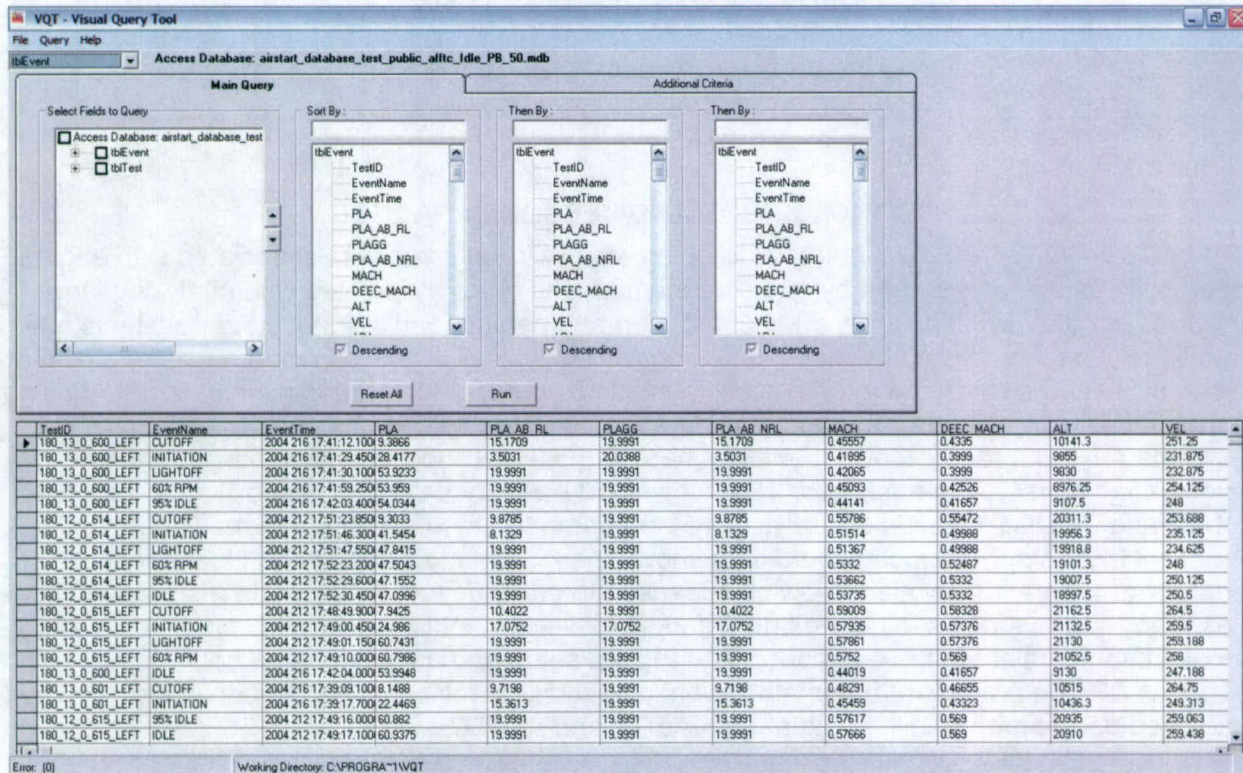


**Table 8 Comparison of Time-to-Idle for Two Different Determinations of Idle**

Test Point	Aim Conditions			Idle at PB = 50	Idle at bit = 1	Time Difference
	Altitude (ft)	Mach or KCAS	N2 (% RPM)	Time to Idle (sec)	Time to Idle (sec)	Time to Idle (sec)
1	Low	Low	30	40.46	40.06	0.40
2			50	17.96	17.96	0.00
3			26	38.6	38.55	0.05
4	Medium	Low	30	79.04	79.39	-0.35
5			50	34.52	34.97	-0.45
6		Medium	30	26.12	26.47	-0.35
7			50	14.3	14.70	-0.40
8		Low		67.94	68.44	-0.50
10				High	21.02	21.47
11	High	Low		57.86	58.26	-0.40
12	Medium		30	51.98	52.28	-0.30
13			50	18.98	19.53	-0.55

## USES OF THE DATABASE

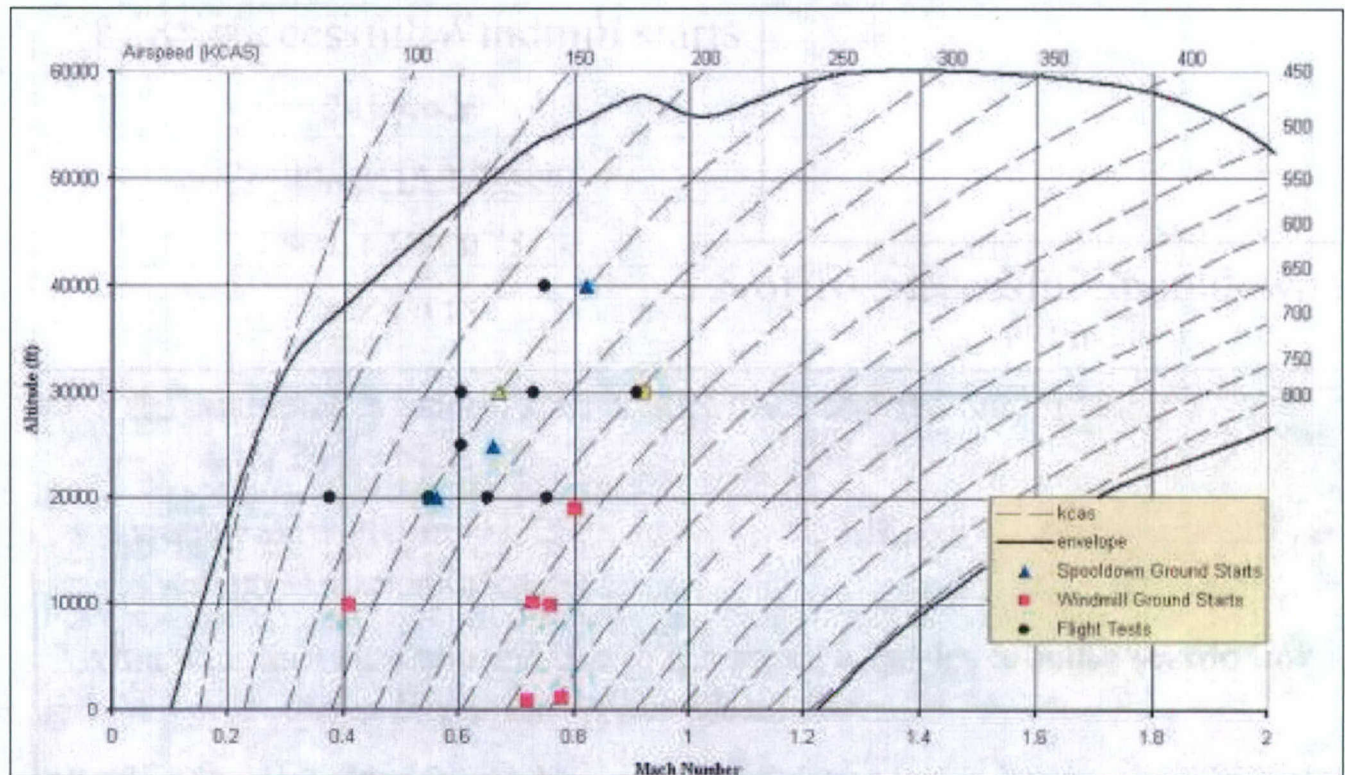
The AirTrans AP has long term usefulness even after data analysis was completed. Once data was sent to the database, it could be sorted and queried. Data from a future test project could be added to the database. Figure 16 presents a Visual Query Tool that could be used to sort and query data.





**Figure 16 Visual Query Tool (VQT) for the AirTrans AP**

Once the database was sorted and queried, the focus became a matter of what to do with the data. An obvious choice was to load the data into tables and plots. However, there are many options. Figure 17 presents one option: a skychart. This is a notional skychart showing a comparison of where airstarts were completed during flight test and during a recent ground test at AEDC.



**Figure 17 Notional Airstart Skychart**

### **CONCLUSIONS AND LESSONS LEARNED**

The AirTrans AP offers great potential to automate the initial processes associated with engine operability data analysis, thereby reducing the time and costs associated with analyzing large volumes of data. Engine starts, augmented, non-augmented, and maneuvering transients have traditionally been analyzed by manual methods. This typically involved a test engineer sitting at a computer and analyzing each point, using multiple approaches for reading in data files, finding and databasing key event information, and generating associated plots and listings. The AirTrans AP has already shown that it can reduce the time associated with many of these engineering duties. The application has shown that its event detection capabilities are very robust and that the products of analysis are very comparable to traditional manual methods. Additionally, the databasing of event data shows the long term value beyond just current analysis requirements, since it allows comparison to previous test results. The AirTrans AP still has many improvements to go through before its development is considered complete. Most of these changes relate to the ease of configuring the setup and data files. The heart of the AirTrans AP, the process of automated event detection, is well on its way to becoming an accepted part of propulsion testing at both AEDC and AFFTC.



## **ACKNOWLEDGMENTS**

I would like to thank David Kidman, Jeff Corn, 2d Lt Chi Lam, and Cadet Tim Killham for assisting me in researching and writing this paper. David Kidman introduced me to the application and helped me to understand the process of engine operability analysis. Jeff Corn helped me to understand how the application works. Lt Lam developed a Visual Basic GUI tool to add a header to data files, which greatly sped up my work. Last, but not least, I would like to thank Cadet Killham from the United States Air Force Academy for his assistance with the manual analysis.

## **BIBLIOGRAPHY**

Corn, Jeff D., *AirTrans Prototype User Guide: Version 0.8*, Air Force Flight Test Center, Edwards AFB, CA, May 2004

Technical Order 1F-15E-1, *Flight Manual, USAF Series: F-15E Aircraft*, Warner Robins AFB, GA, January 2004

## **BIOGRAPHY**

1st Lt Khoi Ta is a graduate of California State Polytechnic University Pomona. He has a Bachelor of Science in Aerospace Engineering. Lt Ta obtained his commission from Air Force Officer Training School (OTS) as a developmental engineer. Before his commission, he was enlisted the U.S. Marine Corps reserves. Edwards AFB is his first duty station as an Air Force officer.